

**Amendments to the Claims:**

The following listing of claims replaces all prior versions, and listings, of claims in the present application. Please amend claims 1, 6 and 9 as follows:

**Listing of the Claims:**

1. (currently amended) A method for use in a computer system that effects secure access to a store, comprising:

receiving a request to access a store from a first process initiated by a requester;

initiating a second process responsive to said store access request, wherein data generated by said second process is accessible to said first process but inaccessible to the requester;

changing a context of said second process to the user id of said store;

providing said store with an exclusive user id, said exclusive user id being different from a user id of the requestor;

said second process receiving tokenized credentials corresponding to the user id and password of the requestor from said first process responsive to said request without the use of files and without interaction with said requestor;

said second process converting the tokenized credentials to the user id and password of the requestor and performing a lookup of said ~~tokenized credentials~~ user id and password of the requestor in a credential store;

said second process passing a user id and password associated with said store to said first process if said ~~tokenized credentials are recognized~~ user id and password of the requestor are found in said credential store;

communicating between said first process and said store via inter-process pipes  
~~private communications channels, said communications channels being inaccessible to the~~  
requester; and

said first process obtaining data from said store via said ~~private communications~~  
~~channels~~ inter-process pipes responsive to said store access request using said user id and  
password associated with said store.

2. (original) A method in accordance with claim 1 further comprising: sending the data  
responsive to said store access request to a downstream process.

3. (original) A method in accordance with claim 1 further comprising: logging said store  
access request.

4. (original) A method in accordance with claim 3 wherein sending the data responsive  
to said store access request to a downstream process further comprises never sending the data to  
the requestor.

5. (original) A method in accordance with claim 1 wherein said computer system  
employs the UNIX operating system and wherein said context changing comprises invoking the  
UNIX set user id facility.

6. (currently amended) A computer system comprising:

a data store having an exclusive user id in the computer system;

a system for providing secure access to said data store, said system being configured to receive a request for access to said data store from a first process initiated by a requester, said exclusive user id being inaccessible to said requester, said exclusive user id being different from a user id of the requestor, said first process being and configured to initiate a second process responsive to said request for access to said data store, wherein data generated by said second process is accessible to said first process but inaccessible to the requester, said second process being responsive to said request for access to said data store by changing its user id to the exclusive user id, said second process being operable to receive tokenized credentials corresponding to the user id and password of the requestor from said first process responsive to said request without the use of files and without interaction with said requester, said second process converting the tokenized credentials to the user id and password of the requestor and being operable to perform a lookup of said ~~tokenized credentials~~ user id and password of the requestor in a credential store, said second process being operable to pass a user id and password associated with said data store to said first process if said ~~tokenized credentials are recognized~~ user id and password of the requestor are found in said credential store; and

inter-process pipes ~~private communications channels~~ between said first process and said data store for communicating and responding to requests for data by said first process using said user id and password associated with said data store, said inter-process pipes ~~communications channels~~ being inaccessible to the requester.

7. (original) A computer system according to claim 6 wherein the system for providing secure access is further configured to provide data from said data store to a downstream process.

8. (original) A computer system according to claim 6 wherein the system for providing secure access is further configured to log requests for access to said data store.

9. (currently amended) A method for use in a computer system that operates under the UNIX operating system that effects secure access to a store, comprising:

executing a shell script which creates a data stream containing a tokenized user id and password in order to initiate a request to access a store from a first process initiated by a requester, said tokenized user id and password corresponding to the user id and password of the requestor;

said first process initiating a second process responsive to said store access request, said second process and said store being in a protected area, wherein data generated by said second process is accessible to said first process but inaccessible to the requester;

said second process obtaining the tokenized user id and password of the requestor from said first process along said data stream;

changing a context of said second process to the effective user id of said store using the UNIX set user id facility, said user id of said store being different from a user id of the requestor;

said second process providing said store with said effective user id, said effective user id being different from said user id of the requester;

said second process receiving said tokenized user id and password from said first process responsive to said request without the use of files and without interaction with said requester;

said second process converting the tokens to said user id and password of the requestor and performing a lookup of said user id and password of the requestor in a credential store;

said second process passing a user id and password associated with said store to said first process along a second data stream if said user id and password of the requestor are found in said credential store;

communicating between said first process and said store via a sending inter-process pipe and a receiving inter-process pipe, said first and second inter-process pipes being inaccessible to the requester; and

said first process obtaining data from said store via said second inter-process pipe responsive to said store access request via said first inter-process pipe using said user id and password associated with said store.

receiving a request to access a store from a first process initiated by a requester;

initiating a second process responsive to said store access request, wherein data generated by said second process is accessible to said first process but inaccessible to the requester;

10. (previously presented) A method in accordance with claim 9 further comprising:  
sending the data responsive to said store access request to a downstream process.

11. (previously presented) A method in accordance with claim 9 further comprising:  
logging said store access request.

12. (previously presented) A method in accordance with claim 11 wherein sending the  
data responsive to said store access request to a downstream process further comprises never  
sending the data to the requestor.